Predicting Poverty levels in South Africa

Jessica van der Berg[a]

[a]*Stellenbosch University, South Africa*

Abstract

South Africa has extremely high levels of poverty for an upper middle-class country. To help reduce the poverty levels, the government has many grants that are available to households that intend to improve the quality of life of poor South African's. However, accurate targeting is challenging due to the lack of income data. Machine learning techniques can improve the distribution of grants by predicting poverty levels. This paper will show that *decision trees* and *random forest* models predict poverty levels with a high degree of accuracy in South Africa.

Keywords: Machine Learning, Poverty Prediction, South Africa.

JEL classification L250, L100

1. Introduction

Since apartheid ended in 1994, the South African Government has committed a significant number of resources to distribute grants effectively and efficiently to the poor and the vulnerable. However, South Africa has remained a country with extremely high levels of inequality and poverty for an upper middle-income country Stats (2011). Previous literature has shown that poverty levels are higher in rural areas than in urban areas. This is largely due to rural household not having access to the same employment opportunities has urban households. South Africa is a country with high levels of unemployment and extremely low wages. This implies that individuals that are economically active can still fall below the upper or lower bound poverty line due to the low wages that they receive Leibbrandt, Woolard, Finn & Argent (2010).

Poverty remains unacceptably high for a country of South Africa's economic status and remains closely associated with race. Thus, poverty reduction remains one of the key economic goals. Poverty in money terms has declined markedly since apartheid ended in 1994. This was made possible by the expansion of the social grant system. However, accurately targeting social welfare programs can be challenging given that income data is often incorrect Yu & Van der Berg (2017). To overcome this problem, households are subject to a proxy means test (PMT) to identify whether a household

Email address: 20190565@sun.ac.za (Jessica van der Berg)

qualifies for social assistant. This essay will try to identify new methods to identify households that are below the food poverty line, lower-bound poverty line and upper bound poverty line using machine learning techniques. After analyzing various different models, the paper concludes that a decision tree and random forest model most accurately predicts poverty.

The paper proceeds as follows. Section 2 presents a brief literature review with respects to poverty in South Africa. Section 3 discuss the data manipulations and analyses descriptive statistics. Section 4 provides a theoretical discussion about the methodology used. Section 5 reports of the results, and finally, section 6 concludes.

2. Literature Review

Social spending has become a major tool for targeting resources to the poor. Since apartheid, the poor and vulnerable get significantly more then their share of social spending. However, poverty levels in South Africa have not improved much. Government has gone to considerable lengths to improve targeting and access for the poor to social services but the underlying reasons for the improvement in targeting are not solely related to good policy and delivery. Social spending is not distributed efficiently which implies that the poor receive limited gains from any grants that they receive Van der Berg & Moses (2012).

SA's post-transition government has enjoyed considerable success in shifting spending to the poor. But the overwhelming message conveyed by the data on social service delivery is that social spending has often not produced the desired social outcomes, both in social delivery programmes, and in households, particularly the most vulnerable. SA urgently needs to strengthen the links between fiscal resource shifts and social outcomes.

In light of the COVID-19 pandemic, social spending is now more important than ever as many individuals have lost their jobs, and many families their homes. The government responded to the pandemic by increasing social spending however there is still uncertainty as to whether social grants are being successfully targeted. Köhler & Bhorat (2020) showed that grants make up a big portion of total monthly household income for poor households and provides a stable income for the most vulnerable. Therefore, the accurate targeting of social programs is extremely important.

Machine learning techniques can help predict poverty, and by extension, improve the targeting of social programs. Impressive working has been done by Jean, Burke, Xie, Davis, Lobell & Ermon (2016) where the researchers use machine learning techniques to estimate consumption expenditure from high-resolution satellite imagery. Blumenstock, Cadamuro & On (2015) successfully used machine learning techniques to predict wealth throughout Rwanda. This paper will focus on evaluating the best machine learning model to predict poverty levels in South Africa. The next section will give an

overview of the data used for this study.

3. Data

The data used for predicting poverty level was exacted from the General Household Survey (GHS) 2018, which is a survey completed annually by Statistics SA to measure the living circumstances of households in South Africa. They survey includes household and individual characteristics. After taking out all of the NA values and taking out households who did not know or answer the relevant questions, the dataset consists of 14 546 entries.

Since the GHS consist of household survey, total income per household is reported. To calculate the average monthly income per individual within each household, I first need to calculate the total number of adults within each household. This is done by taking the difference between household size and the number of children under the age of 17. I then take the total monthly income and divide it by the total number of adults to get the average monthly income per individual in each household. This income information is then used to divided the data into four groups.

The first group is individuals whose income fall below the food poverty line. In 2018, the food poverty line was R547 per month. The food poverty line is also referred to as the extreme poverty line as it refers to the absolute minimum amount an individual will need to be able to afford the minimum energy intake for survival. The second group consist of individuals whose monthly income falls between the food poverty line and the lower-bound poverty line (R785). The lower bound poverty line is the sum of the food poverty line and and minimum amount for non-food items. The third group consist of individuals between the lower-bound poverty line and the upper-bound poverty line (R1183). The final group consist of individuals whose monthly income is above the upper-bound poverty line, which I refer to as non-vulnerable individuals. Figure 1 graphically displays the process described above in the format of a decision tree, where 1 represents the food poverty line, 2 the lower-bound poverty line, 3 the upper-bound poverty line and 4 the non-vulnerable.

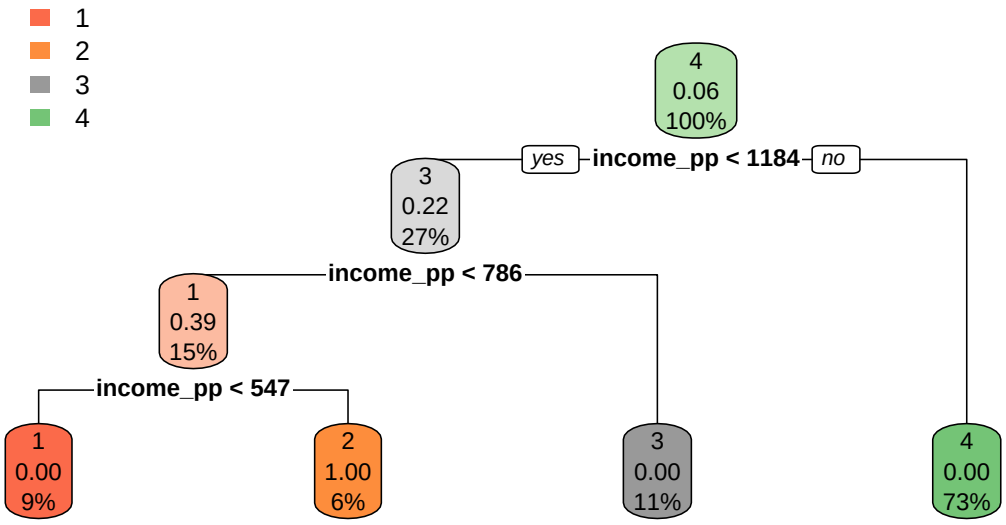

Figure 3.1: Decision tree for poverty levels

After all data cleaning was done, 73 percent of households were non-vulnerable, 11 percent were between the upper and lower-bound poverty line, 6 percent between the lower and food poverty line, and 9 percent fell below the food poverty line. To evaluate the performance for the machine learning techniques that I will implement, I randomly split the data into two subsets using a 70:30 ratio. The training dataset will consist of 70 percent of the original dataset, while the test dataset will consist of the remaining 30 percent. Figure 2 below shows the number of household per poverty level for the training dataset.

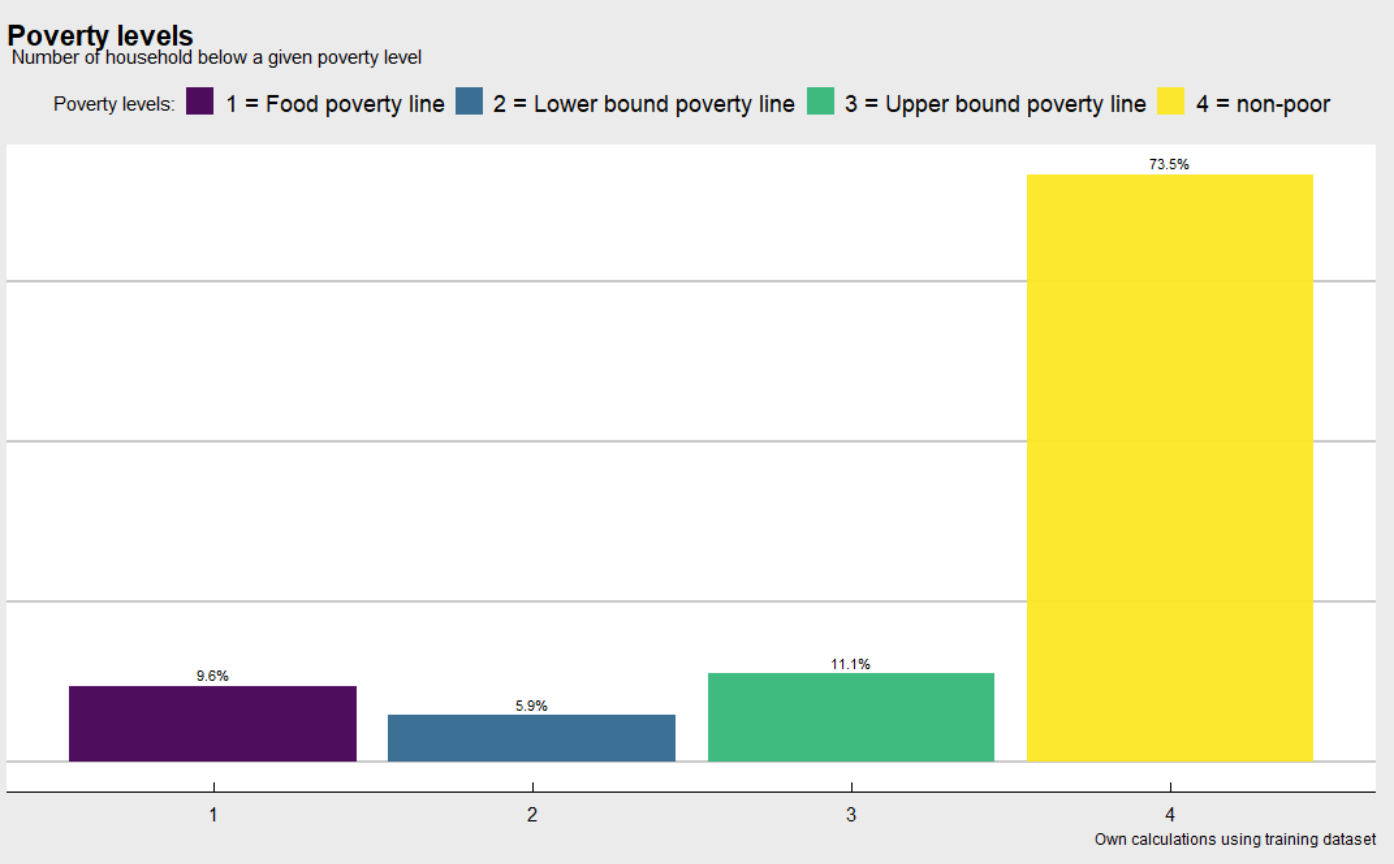

Figure 3.2: Number of household per poverty level

Figure 2 shows that in our training dataset, 73.5 percent of households are non-vulnerable and 9.6 percent of households monthly income falls below the food poverty line. This implies that I have an extremely imbalanced dataset. This is important as it will have an effect on the machine learning techniques that I implement later on.

Furthermore, I analyze some of the variable in more details. From the density plots in figure 3.3, it shows that households whose average income per person falls below the upper-bound poverty line (green line) tend to have slightly older head of household (\emph{head_age}) then those household belonging to other poverty levels. A household size (*hholdsz*) of 1-2 individuals tend have a larger probability to be non-vulnerable (yellow line), where larger household size of 3-5 individuals then have a larger probability of falling below the food poverty line. This implies that smaller households are more well-off than larger households. Total expenditure (*Q814Exp*) is higher for the non-vulnerable households, which makes sense as they receive a higher income and therefore have more money to spend. The total number of rooms (*Q55TotRm*) seem to be distributed similarly, with non-vulnerable households having a slightly smaller distribution. This comes as no surprise as these non-vulnerable households tend to have a smaller household size, implying that they need less bedrooms. Furthermore,

non-vulnerable household have much larger property valuation (*Q58Val*), which could imply that they have a higher standard of living.

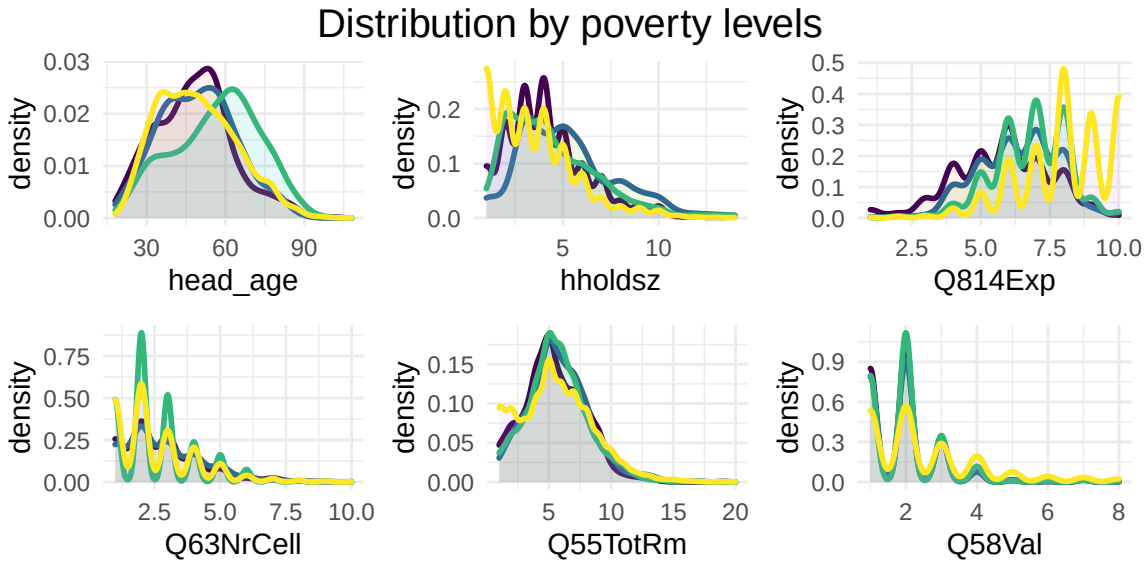

Figure 3.3: Distribution of certain variables by poverty level

Before we start building models I also analysis which variables are correlated with the dependent variable (which is is poverty level). Figure 3.4 displays the top 10 variables that are correlated with the dependent variable. Our top four that are positively correlated are *econact hh*, which is a binary variable indicating whether or not the household is economically active, *Q814Exp*, which is a numeric variable display total household expenditure, *totmhinc*, which is a numeric variable displaying total household income, and *Q89aGrant*, which is a binary variable indicating whether the household receives a grant or not.

Variables that are negatively correlated with the dependent variable is the sex of the head of the household (*head sex*), whether or not the household owns a washing machine (*Q821WashM*), the household size (*hholdsz*), the amount spent on rent or mortgage of property per month (*Q57Rent*) and the number of adults in the household (*adults*).

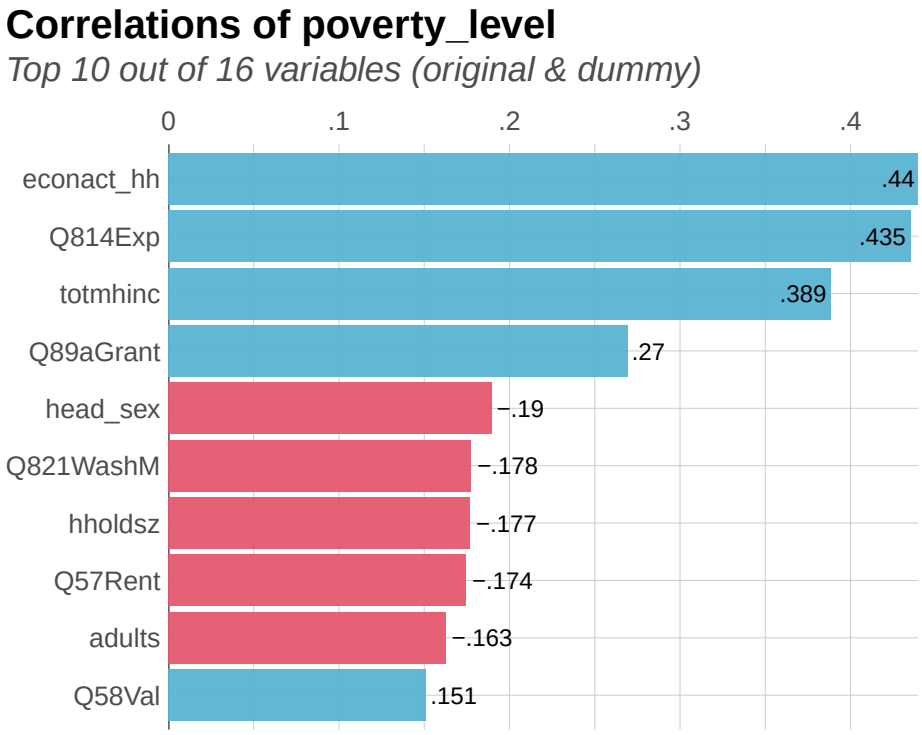

Figure 3.4: Correlation with dependent variable

4. Methodology

4.1. Analysing Classification Models

To see which model predicts poverty most accurately and efficiently, I assess the performance of seven supervised classification models using the caret package. The dataset is extremely unbalanced; therefore, the defaults parameter will be Kappa to order to improve the performance of the models. The Kappa metric compares the observed accuracy with the expected accuracy. It also accounts for random chance which implies that it makes the model more accurate that simply using an Accuracy as a metric. The Kappa matric is calculated using the formula;

$$k = \frac{p_0 - p_e}{1 - p_e}$$

Where p_0 represents the overall accuracy of the model and p_e represents that measure of the agreement between the predictions and actual class value of the model. Therefore, Kappa attempts to account for evaluation bias by considering the correct classification by a random guess Dalpiaz (2017).

The table below briefly discusses the seven different classification models that I compared to determine which model is the most accurate to predict poverty Boehmke & Greenwell (2019).

Model	Reference Name	Description
Multinomial Logistic Regression	multinom	Makes use of the maximum likelihood estimation to evaluate the probability of a categorical relationship
Linear Discriminant Analysis	lda	Used to find linear combinations of separates multiple classes of features, representing the dependent variable as a linear combination of other features
Naive Bayes	naive bayes	Using Bayes Theorem, this model applies posterior probability to the categorization, making the uneducated assumption that the predictors are independent
Linear Support Vector Machine	svmLinear	The model creates a line that separates data into classes
K-Nearest Neighbor	knn	Each observation in the dataset is predicted based on its similarity to other observations
Recursive Partitioning	rpart	Builds models using a general structure which consist of a two-stage procedure and then final presenting the model as a binary tree
Ranger	ranger	An updated and fast implementation of random forest for big data

4.1.1. Evaluation

To ensure that the correct model is chosen, the accuracy of each model is evaluated and compared. However, since the dataset is unbalance, I also analyze the F measure, also known as the F_1 score, of each model. The F_1 score communicates the average between the precision and the recall of each model. A perfect model has an F_1 score equal to 1, therefore models with a higher F_1 score is preferred over models with a lower F_1 score. The formula for the F_1 score is given below;

$$F_1 = 2 \times \frac{precision \times recall}{precision + recall}$$

The extremely unbalanced data will affect the results of the F_1 score. Therefore, it is also informative to evaluate the macro F_1 score and the weighted F_1 score. The macro F_1 is not affected by unbalanced data and is equal to the average of the F_1 score and is commonly used when there are multiple levels or classes. It gives the same importance to each poverty level. A higher macro F_1 score is preferred to a lower one. The formula for the macro F_1 score if given below;

$$Macro\, F_1\, score = \frac{1}{N}\sum\nolimits_{i=0}^{N} F_1$$

Where N is the number of different poverty levels and i is the levels index. The drawback of the macro F_1 measure is that is gives equal weight to all poverty levels, which implies that it over emphasis the under-represented poverty levels. The weighted average F_1 score is similar to the macro F_1 score, but here the F_1 score is weighed according of the number of households from the specific poverty level, which emphasis poverty levels according to size of each poverty level. The formula for the weighted F_1 score is given below;

$$Weighted\, F_1\, score = \frac{n_i \sum_{i=1}^{k} F_1}{\sum_{i=n}^{k} n_i}$$

4.2. Decision Tree

As I will show in section 5.1, decision trees have the fastest computational time without having to compromise much on the accuracy of a model. Decision trees are constructed through an algorithmic approach that identifies the most optimal way to split a dataset based on the information in the dataset. Decision trees are displayed in a flowchart-like structure where each internal node represents some sort of test on a specific feature. Each leaf node then represents a poverty level. The path from the root (the first node) to the leaf represents the classification rules. Decision trees are relatively easy to interpret and therefore, are commonly used Boehmke & Greenwell (2019).

4.3. Random Forest

In section 5.1, I also show that random forest provides perfect accuracy out of all the classification models that are considered, however, it also has the longest computational time. Random forest uses multiple decision trees to provide more flexibility and better accuracy, while reaching a single result. Random forest searches for the best feature form a random subset of features which leads to it providing more randomness to the model. The increased in randomness is what improves the model accuracy as it ensures a low correlation among the multiple decision trees Breiman (2015).

5. Results and Discussion

5.1. Classification Models

The table below shows the different metrics to make comparing different models easier. The first feature that is observed is that the models vary drastically in time. The *lda*, *rpart* and *naive bayes* models are extremely fast whereas *multinom* and *ranger* take relatively long to run. Furthermore, *ranger* scores the best for accuracy and F_1 measures.

Models	Time	Accuracy	Macro F_1	weighted F_1
multinom	37.17	0.9935839	0.9820473	0.9935798
lda	0.81	0.7774977	0.4561088	0.7409658
naive bayes	1.67	0.8819890	0.8138291	0.8862441
svmLinear	9.91	0.9660862	0.9134729	0.9661192
knn	4.81	0.9869386	0.9714937	0.9876236
rpart	1.08	0.9372136	0.9155884	0.9137702
ranger	41.92	1.00000000	1.00000000	1.00000000

Analyzing the accuracy score, Figure 5.1 shows that *ranger* has the highest accuracy, followed closely by *multinom*. However, these two models also take the longest to run. *knn* and *rpart* are relatively fast and they have a high level of accuracy. For the fastest running models, decision trees (*rpart*) are the most accurate. Figure 5.1 also implies that there is some trade-off between accuracy and computational time.

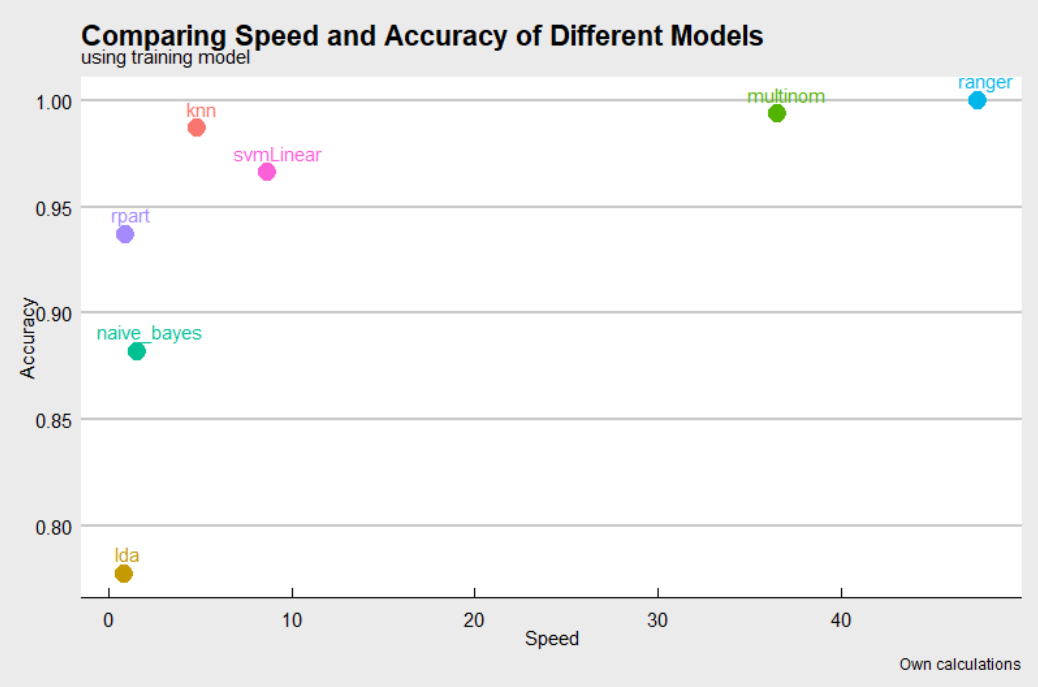

Figure 5.1: Speed versus Accuracy

Figure 5.2 shows the relationship between computational time and the Macro F_1 score measure. *ranger* and *multinom* still have the highest degree of accuracy. However, *rpart* has a higher degree of accuracy while maintain the same computational time. *naive bayes* is performing worst in terms of accuracy, while *knn*, *svmLinear* and *lda* are all performing relatively the same. Here, the same conclusion is

reached as with Figure 5.1, that decision trees (*rpart*) is the most accurate when analyzing models with the fastest computational time.

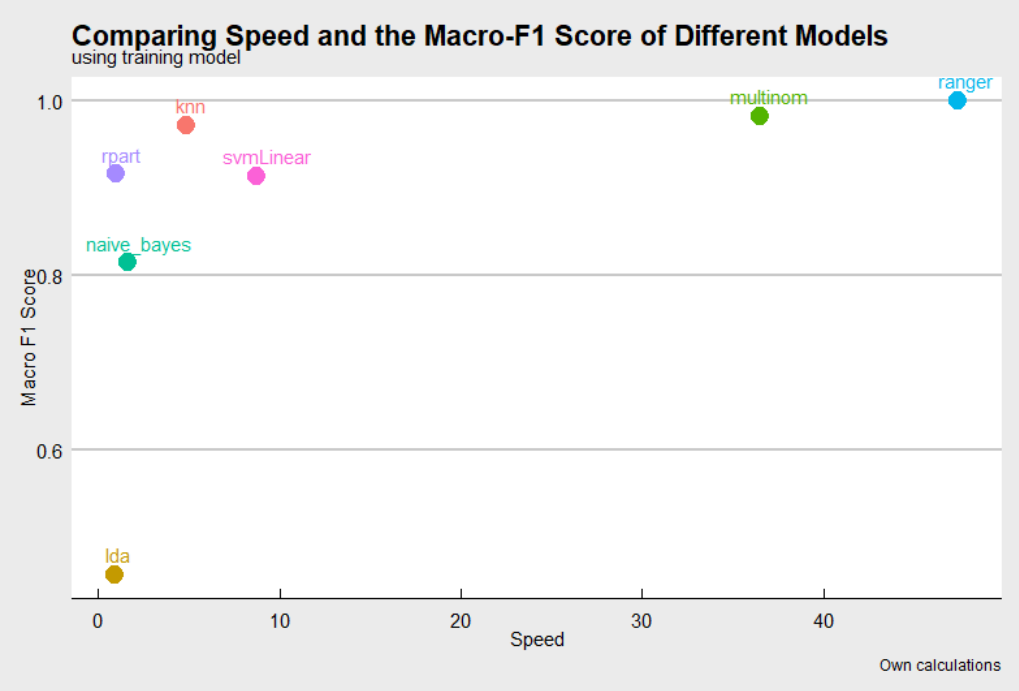

Figure 5.2: Speed versus Macro F1

Figure 5.3 displays that *naive bayes* performs slightly better in terms of accuracy when compared to the macro F_1 score whereas the rest of the models perform similarly. This suggest that random forest (*ranger*) is the most accurate when predicting poverty across all the measures, however, the computational time is extensive. If you are willing to compromise on the accuracy of a model, then decision tree (*rpart*) is the best model, which serves a high degree of accuracy and an extremely fast computational time. Now that I have determined which models are the best to predict poverty, decisions tree and random forest will be analyzed.

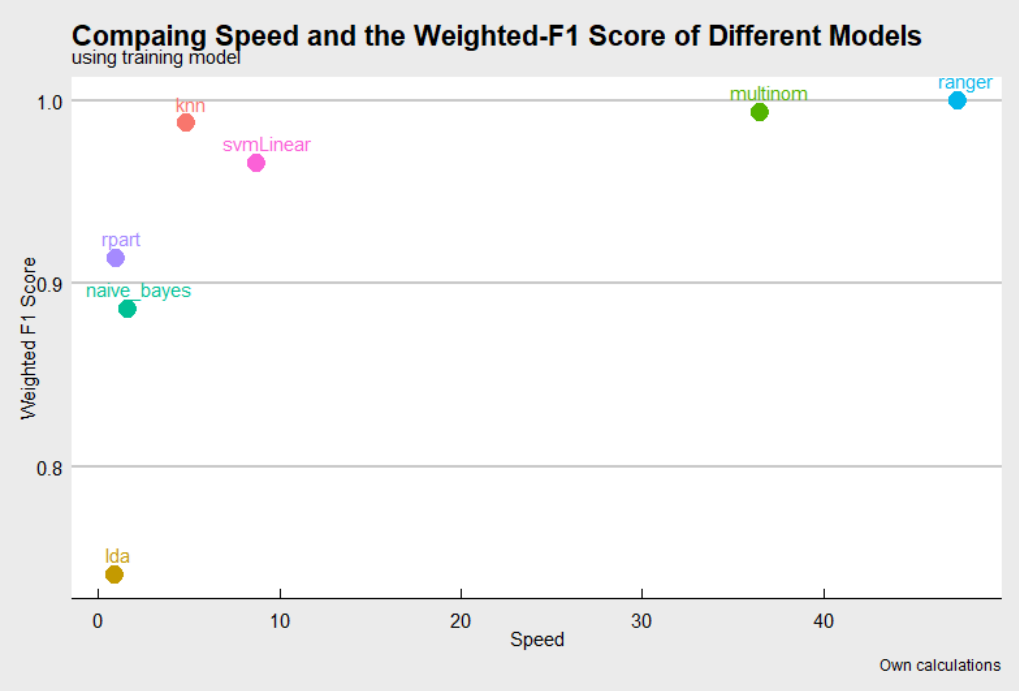

Figure 5.3: Speed versus Weighted F1

5.2. Decision Tree

The decision tree in Figure 5.4 shows the path of classification rules that determine under which poverty level a household is classified, where red (1) represents household falling beneath the food poverty line, orange (2) represents households that are above the food poverty line but below the lower-bound poverty line, purple (3) represents households that are above the lower-bound poverty line but below the upper-bound poverty line and green (4) represents households that are above the upper-bound poverty line.

As we can see, household income (*totmhinc*), the number of adults in the household (*adults*) and monthly salary (\emph{Q42Msal_hh}) are the only variables used to determine the poverty level of a household. The difference between monthly salary and total household income is that total household income consists of wages/salary, grants and any other type of income a household might receive, whereas monthly salary only consists of money received through employment. Figure 5.4 communicates that income variables and the number of adults are the most important variables when determining the poverty level under which households fall.

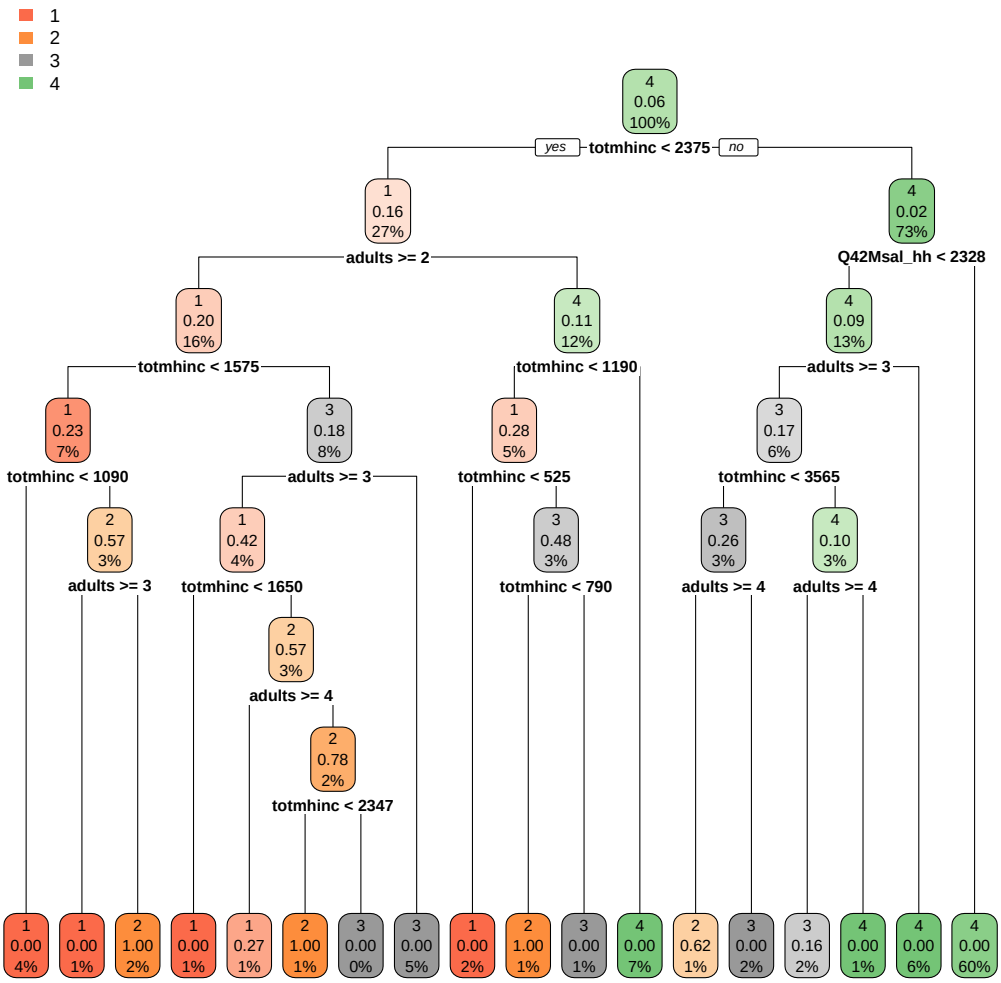

Figure 5.4: Decision Tree

If a household's total monthly income is below R2 375, then there is only a 7 percent chance that the household will earn enough to be above the upper-bound poverty line, and this is only for households that consist of only one adult. Therefore, a single adult can earn between R 1190 and R2375 and still be above the upper-bound poverty line. Households of more than 4 adults tend to be poorer than smaller households. This statement supports the findings of @ lanjouw1995poverty that larger households are more likely to be poorer in developing countries. To further assess which variables, determine under which poverty level a household falls, I construct a decision tree not considering any income variables. This is displayed below in Figure 5.5.

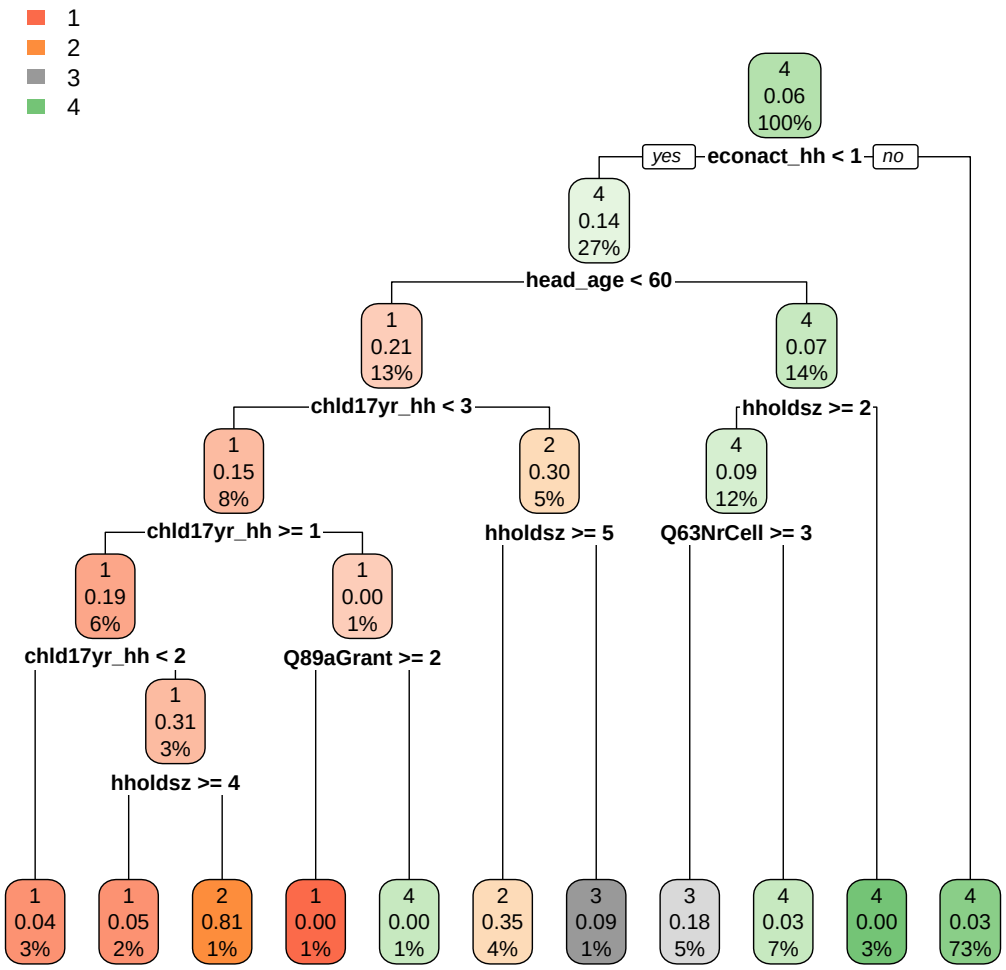

Figure 5.5: Decision Tree excluding income variables

Now that income variables are excluded, a different picture of variables affecting poverty levels are displayed. The first thing we noticed is that all households are economically active are above the upper-bound poverty level. This means the employment is a big determinant of the poverty level a household falls under and that creating more jobs can help reduce poverty.

Furthermore, households that have a head of younger than 60 years of age, tend to be better off than households where the head is older than 60 years. This implies that old age grants, such as pension grants, could be very important source of income to poorer households. Pension grants provide a regular income to poor households, meaning that it has the potential to reduce poverty. Duflo (2003) found that pension grants also have a positive effect on the nutrition and health of young girls. Therefore, pension grants can reduce poverty as well as improve the health on children. It also seems that poorer households receive more than two grants, however, the grants fail to keep the household

above the food-poverty line.

5.3. Random Forest

The default random forest performs 500 trees, and the number of variables tried at each split are 4. Averaging across all the trees, the out-of-bag (OOB) error rate, which measures the prediction error of the random forest model, is equal to 1.27 percent. The classification error for the food poverty line is 2.7 percent, for the lower-bound poverty is 8.3 percent, for the upper-bound poverty line is 3.8 percent, and for the non-vulnerable is 0.12 percent.

Figure 5.6 plots the error rate across the numerous amounts of decision trees; thus, we can find which number of trees provide the lowest error rate. AS the number of trees increase, the error rate stabilizes. After 40 decision trees, the error rate seems to stay relatively constant without any significant change.

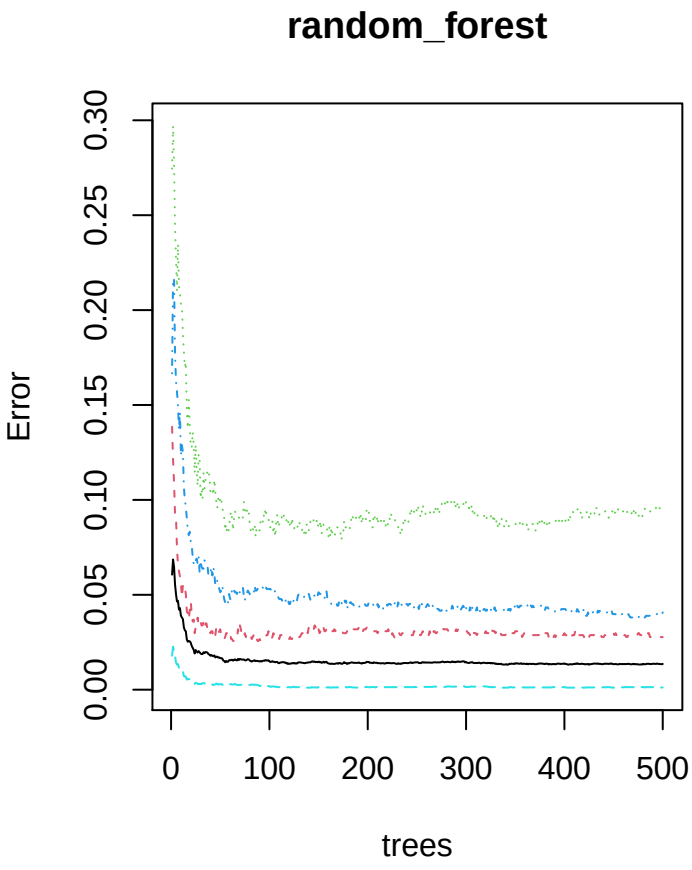

Figure 5.6: Mean square error of forest

5.4. Comparing important variables

The table below compares the most important features of the *rpart* and the *multinom*. The *multinom* models Is used since it is the only other model that the *varImp* function will accept. For the *rpart* model, the most important variables come as no surprise. The income variables are the most important, followed by whether the head of the household is economically active and then household expenditure. We have evaluated these the 4 most important variables here when we analysed decision trees in section 5.2. The most important variables for *multinom* varies greatly from *rpart*, however these variables have also been discussed in section 5.2, where I took out all the income variables from the dataset. What is interesting is that the *multinom* does not consider any of the income variables to be important, as here they have not been removed from the dataset. This results is especially interesting as *multinom* is a close second for the best performing model in terms of accuracy.

Table 5.1: Most important variables, 'rpart' vs. 'multinom'

Variable	Overall	Variable	Overall
income_pp	100.00000	adults	100.000000
totmhinc	43.11656	hholdsz	50.561742
Q42Msal_hh	34.42951	chld17yr_hh	49.433191
econact_hh	21.29235	econact_hh	2.515908
Q814Exp	12.80371	Q89aGrant	1.628790

6. Conclusion

South Africa has extremely high levels of poverty, therefore accurate targeting of social programs is very important. Machine learning techniques presents new ways to analyze data and can help improve the targeting of social programs, and by extension, possibly reduce poverty. This paper attempted to analyze which classification machine learning model best predict poverty levels in South Africa. Seven classification machine learning models were considered in total, and each evaluated and compared using various version of the F_1 metrics. Decision tree provides a fairly accurate model to predict poverty with an extremely fast computational time. On the other hand, Random Forest has perfect accuracy, but the computational time is almost 30 times that of the decision tree. Therefore, when predicting poverty, there is some compromise between computational time and accuracy. The paper concludes that machine learning techniques can successfully help predict poverty.

References

10 Blumenstock, J., Cadamuro, G. & On, R. 2015. Predicting poverty and wealth from mobile phone metadata. *Science*. 350(6264):1073–1076.

Boehmke, B. & Greenwell, B.M. 2019. *Hands-on machine learning with r*. CRC Press.

Breiman, L. 2015. Random forests leo breiman and adele cutler. *Random Forests-Classification Description*.

Dalpiaz, D. 2017.

Duflo, E. 2003. Grandmothers and granddaughters: Old-age pensions and intrahousehold allocation in south africa. *The World Bank Economic Review*. 17(1):1–25.

Jean, N., Burke, M., Xie, M., Davis, W.M., Lobell, D.B. & Ermon, S. 2016. Combining satellite imagery and machine learning to predict poverty. *Science*. 353(6301):790–794.

Köhler, T. & Bhorat, H. 2020. COVID-19, social protection and the labour market in south africa: Are social grants being targeted at the most vulnerable?

Leibbrandt, M., Woolard, I., Finn, A. & Argent, J. 2010. Trends in south african income distribution and poverty since the fall of apartheid.

Stats, S. 2011. Social profile of vulnerable groups in south africa 2002-2010. *Pretoria: Government Printer*.

Van der Berg, S. & Moses, E. 2012. How better targeting of social spending affects social delivery in south africa. *Development Southern Africa*. 29(1):127–139.

Yu, D. & Van der Berg, S. 2017. South african poverty: The current situation and trends since the transition to democracy.